



Бублик В.В.

---

# ПРАГМАТИЧНЕ ПРОГРАМУВАННЯ

доповідь на заключному засіданні Днів науки НаУКМА  
5 лютого 2016 р.



# ПРОГРАМУВАННЯ – що це?

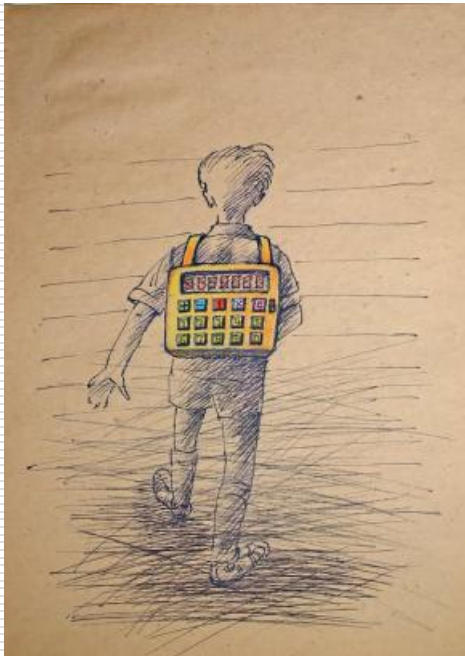
---

- Програмування — процес проектування, написання, тестування, зневадження і підтримки комп'ютерних програм. Програмування поєднує в собі елементи інженерії (існує навіть відповідна спеціальна галузь інженерії — програмна інженерія (англ. software engineering), фундаментальних наук (перш за все математики)



# Програмування – друга грамотність

---



- А.П.Єршов: доповідь на відкритті III Всесвітньої конференції з питань застосування комп'ютерів у освіті (Лозанна, 27-31 липня 1981 р.)

# Програмування як метод

---

Людина незмірно посилить свій інтелект, якщо вважатиме частиною своєї натури здатність планувати свої дії, винаходити загальні правила і спосіб їх застосування до конкретної ситуації, організовувати ці правила в усвідомлену і виразну структуру, - одним словом, **зробиться програмістом**



А.П.Єршов  
1931-1988



# Наука чи ремесло?

---



- *Програміст повинен володіти здатністю першокласного математика до абстракції і логічного мислення в поєднанні з едісонівським талантом до створення чого завгодно з нуля і одиниці*

Знову А.П.Єршов

«Людський фактор в програмуванні»,  
Об'єднана обчислювальна конференція  
Американської федерації товариств обробки інформації,  
18 травня 1972 року, Атлантик-Сіті

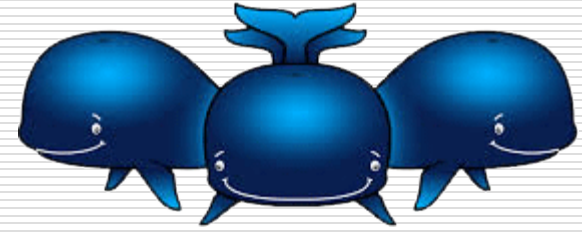
# Програмування як суспільна необхідність

Життєдіяльність нашої цивілізації залежить від програмного забезпечення. Покращення програмного забезпечення і пошук нових сфер його застосування дозволять покращити життя багатьох людей. Програмування відіграє вирішальну роль у цих процесах.



# Джерело – семіотика

---



Чарльз Моріс  
(1901-1979)

- Синтаксис - відношення знаків між собою;
- семантика - відношення знаку до об'єкта;
- прагматика - відношення знаку до суб'єкта

*Foundations of the Theory of Signs*  
(1938)

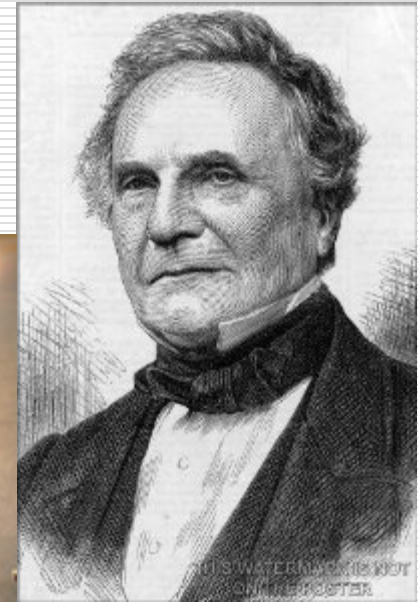
# Предтеча програмування

---



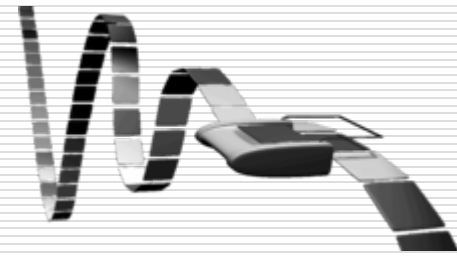
Ada Byron  
Countess of  
Lovelace  
(1815 – 1852)

0,5 %



Charles Babbage  
1792 - 1871



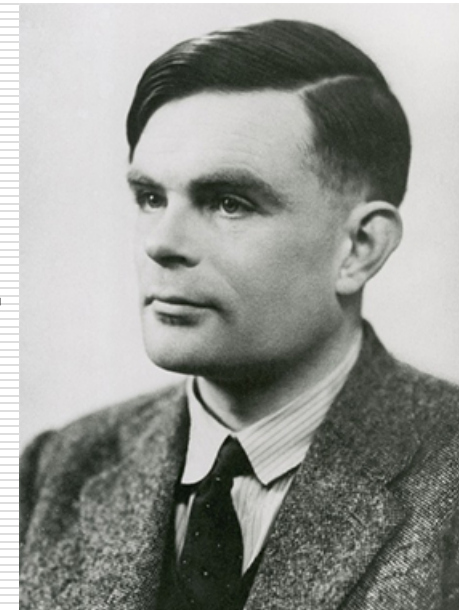


# Прагматика в дії

---

## Друга світова війна

- ❑ 1941 Z3 Конрада Цузе
- ❑ 1942 ABC Джона Вінсента Атанасова і Кліффорда Беррі
- ❑ 1943 Колос Алана Тьюрінга
- ❑ 1950 ENIAC за участю Джона фон Неймана
- ❑ 1950 МЕСМ С.О.Лебедева



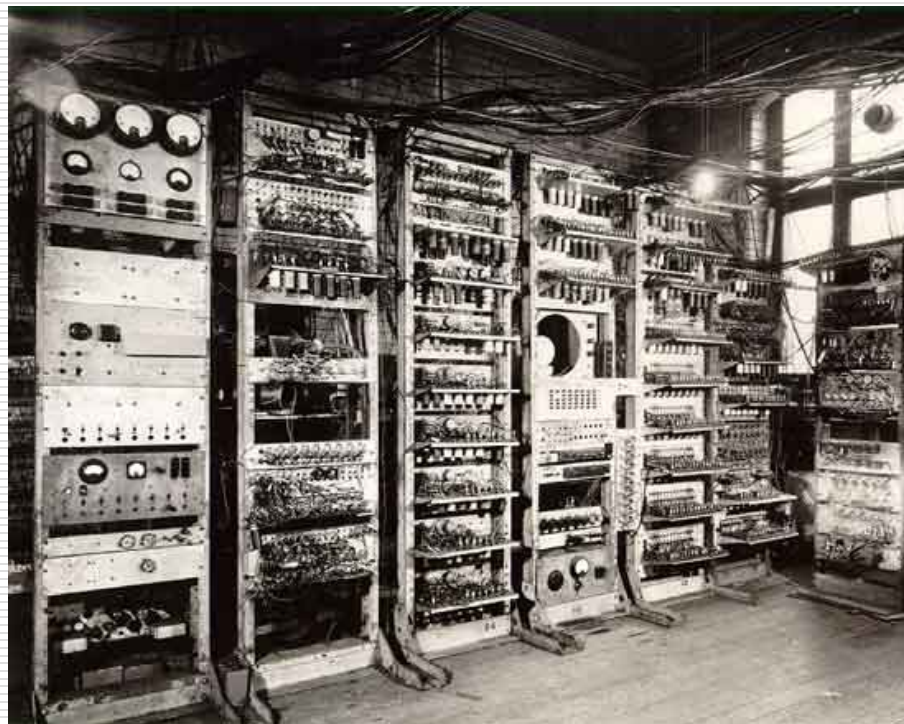
Алан Тьюрінг  
(1912-1954)

# Український вклад

---



С.О.Лебедєв  
1902 - 1974



МЕСМ  
Київ, Феофанія, 1950

# Перша скрипка – синтаксис

❑ FORMula TRANslator =  
FORTRAN (1953)

❑ load A

❑ add B

❑ mult C

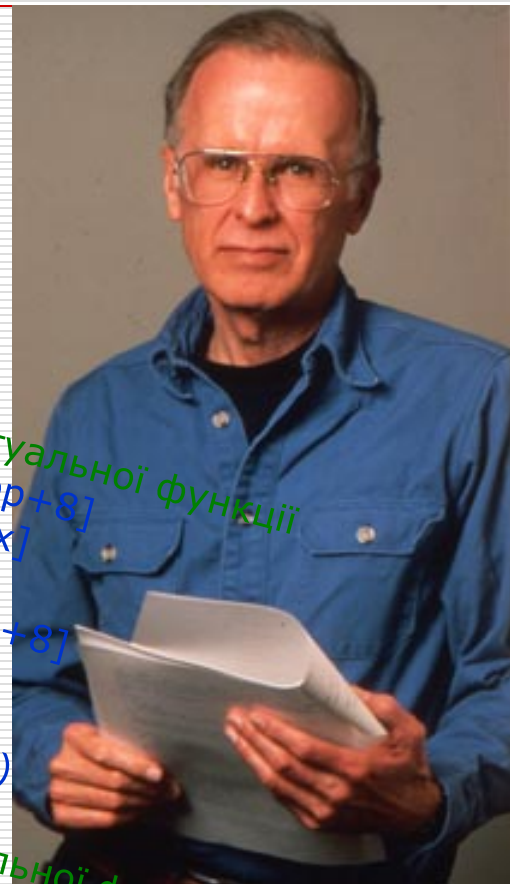
❑ store D

❑  $D = (A + B) * C$

x.f();  
00401207 mov //  
0040120A mov  
0040120C mov  
0040120E mov  
00401211 call  
00401213 cmp  
00401215 call  
0040121A  
x.g();  
00401700 mov //  
00401703 call  
00401708

виклик віртуальної функції  
edx,dword ptr [ebp+8]  
eax,dword ptr [edx]  
esi,esp  
ecx,dword ptr [ebp+8]  
dword ptr [eax]  
esi,esp  
—chkesp (00409620)

виклик невіртуальної функції  
ecx,dword ptr [ebp+8]  
@ILT+805(Base::g) (0040132a)



Джон Бекус (1924-2007)

# Структурна лінгвістика

---



Но́ам Хомський  
(нар. 1928)

Класифікація формальних мов Хомського:

- (3) регулярні;
- (2) контекстно-вільні;
- (1) контекстно-залежні;
- (0) необмежені.

*Logical Structure of Linguistic Theory, 1955*

*Syntactic Structures, 1957*

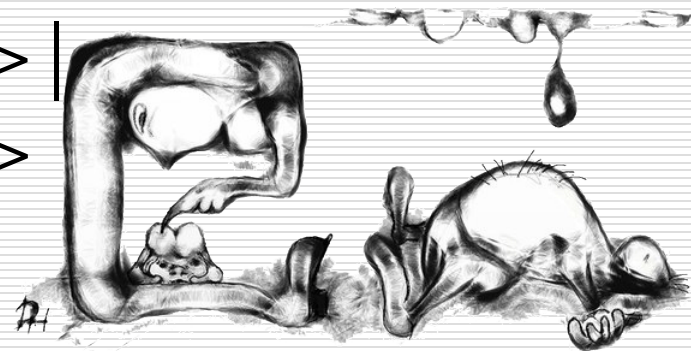


# Нормальні форми Бекуса (БНФ)

---

□ <просте речення> :: =  
<група підмета> <група присудка>

□ < група підмета > :: =  
<підмет> <означення> |  
<означення> <підмет>



□ Глока куздра штеко будланула  
бокра і курдячить бокреня

# Криза синтаксису

---

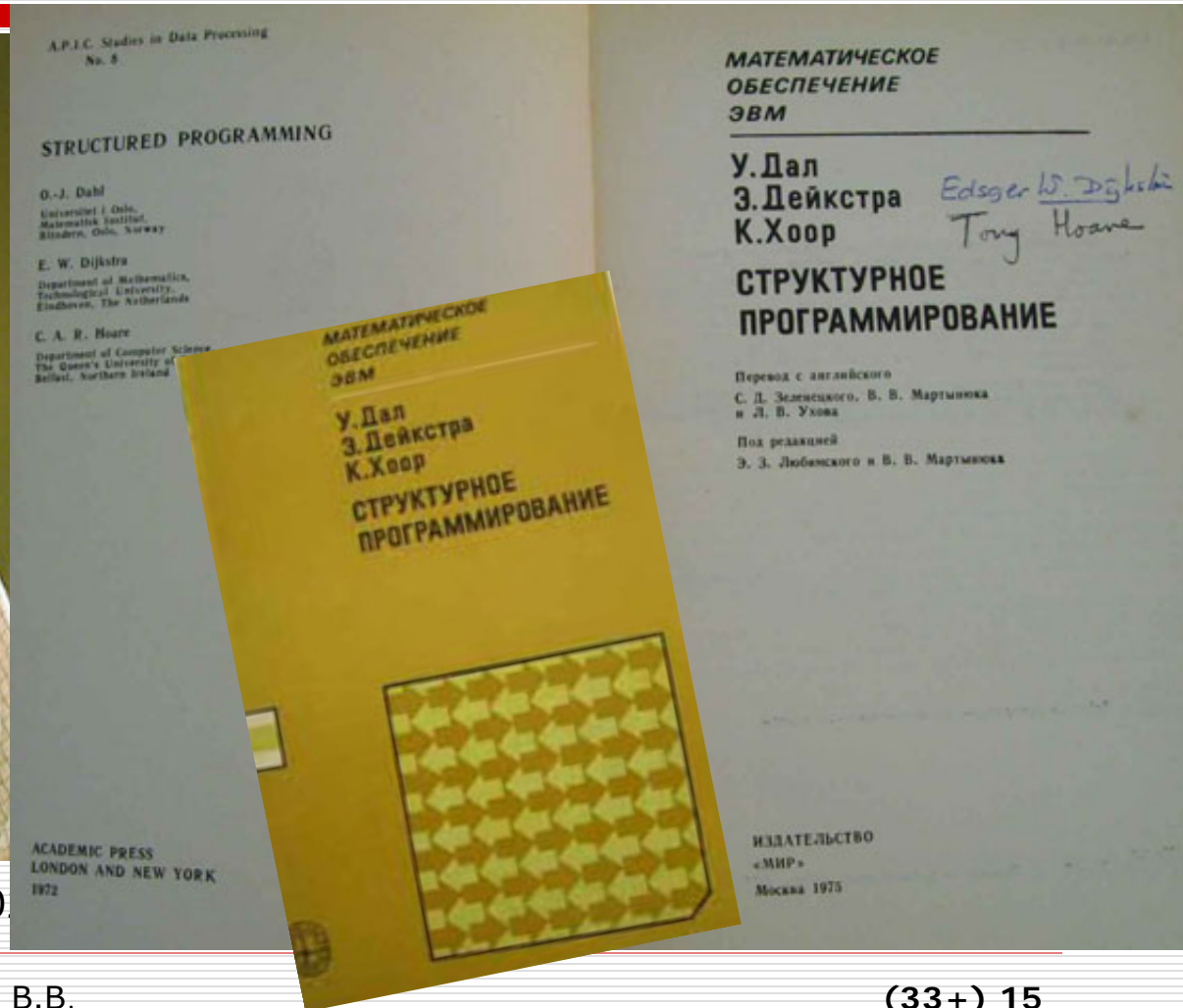
- ❑ правильний оператор  
DO 50 I = 12,525
- ❑ оператор з помилкою  
DO 50 I = 12.525
- ❑ Наслідок помилки:  
невдалий запуск 22 липня 1962  
першого американського супутника  
Венери «Mariner 1»



# Структуроване програмування



Едсгер Дейкстра (1930-2002)



# Обґрунтування структурованого програмування

---

- Теорема про регуляризацію  
(зведення до структурованого  
вигляду) довільного алгоритму –  
В.М. Глушков, 1961, «Успехи  
математических наук»
- Теорема Бьома-Якопіні,  
1966

Коррадо Бьом  
(нар. 1923)



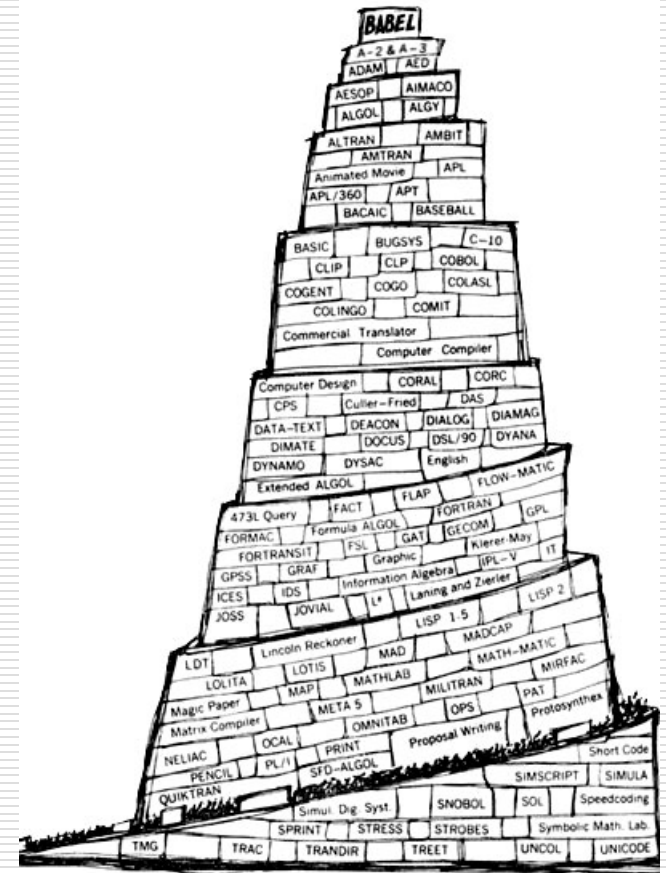
В.М. Глушков  
с сотрудниками  
В.В. Бубликом и  
О.Д. Фелижанко.  
1971 г.





# Лавина мов програмування

- ❑ АЛГОЛ 60
- ❑ ПЛ/1 1964
- ❑ Паскаль 1970
- ❑ С 1972
- ❑ С++ 1983
- ❑ Java 1995
- ❑ С# 2001
- ❑ .....



# Епоха семантики

---

- АЛГОЛ 60
- **ПЛ/1 1964**
- Паскаль 1970
- С 1972
- С++ 1983
- Java 1995
- С# 2001

Віденський метод  
визначення семантики,  
Віденська лабораторія IBM,  
1972



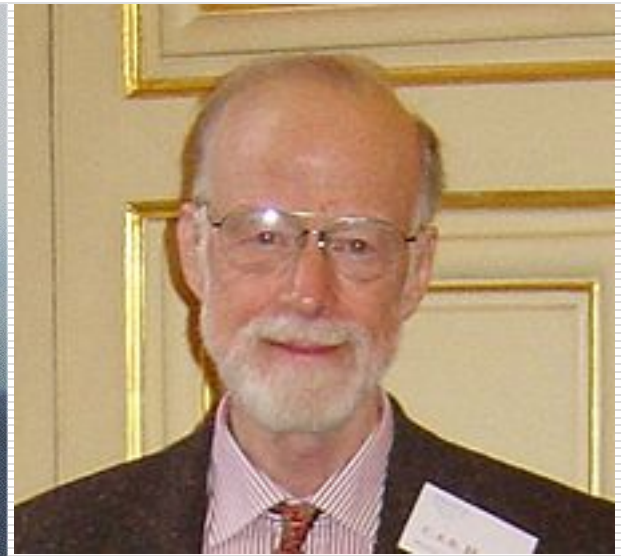
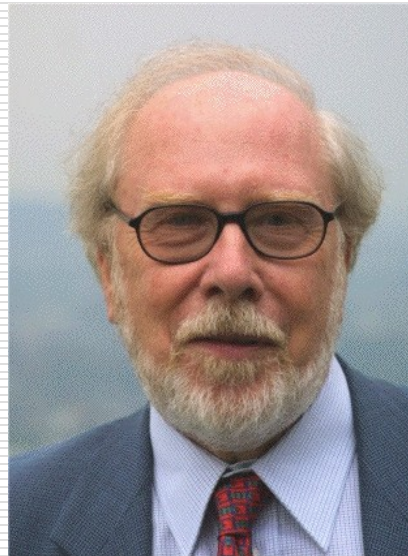
# Епоха семантики

---

- АЛГОЛ 60
- ПЛ/1 1964
- **Паскаль 1970**
- С 1972
- С++ 1983
- Java 1995
- С# 2001

Аксиоматичний метод визначення мови програмування Паскаль

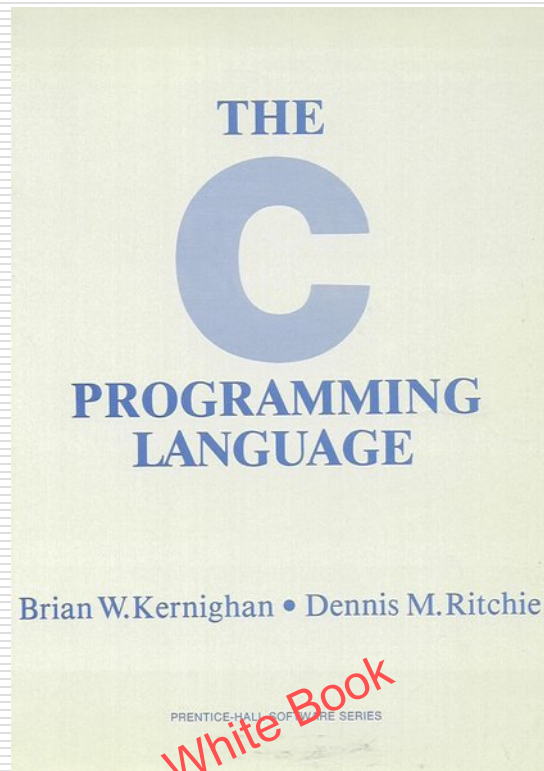
*C. A. R. Hoare, Niklaus Wirth: An Axiomatic Definition of the Programming Language Pascal. 335–355, Acta Informatica, Volume 2, 1973.*



# Знову прагматика

---

- ❑ АЛГОЛ 60
- ❑ ПЛ/1 1964
- ❑ Паскаль 1970
- ❑ **C 1972**
- ❑ C++ 1983
- ❑ Java 1995
- ❑ C# 2001



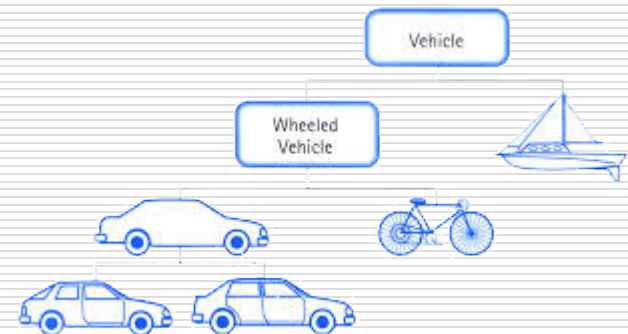
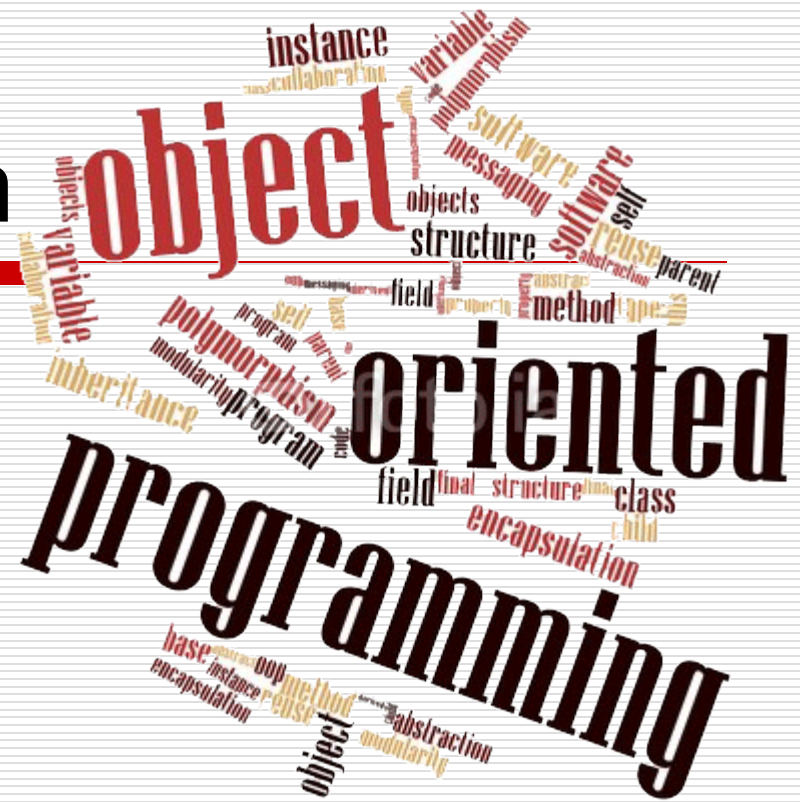
Деніс Річі  
(1941-2011)

Back to hardware!



# Знову прагматика

- ❑ АЛГОЛ 60
- ❑ ПЛ/1 1964
- ❑ Паскаль 1970
- ❑ С 1972
- ❑ **С++ 1983**
- ❑ **Java 1995**
- ❑ **С# 2001**
- ❑ **Python 2008...**



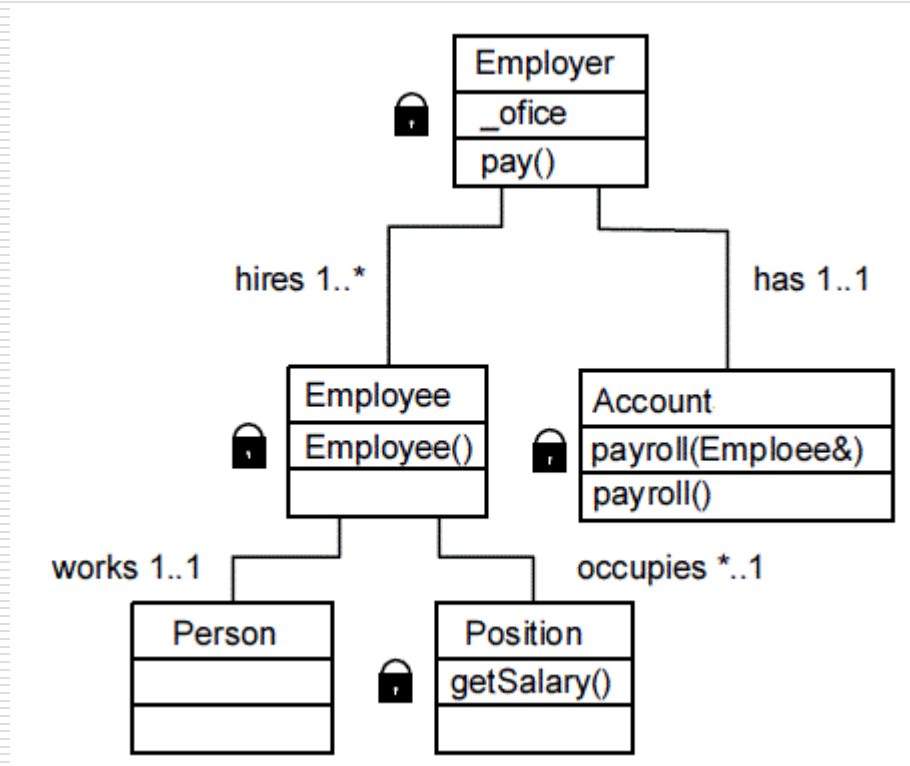
# C++ = C + SIMULA 67

---

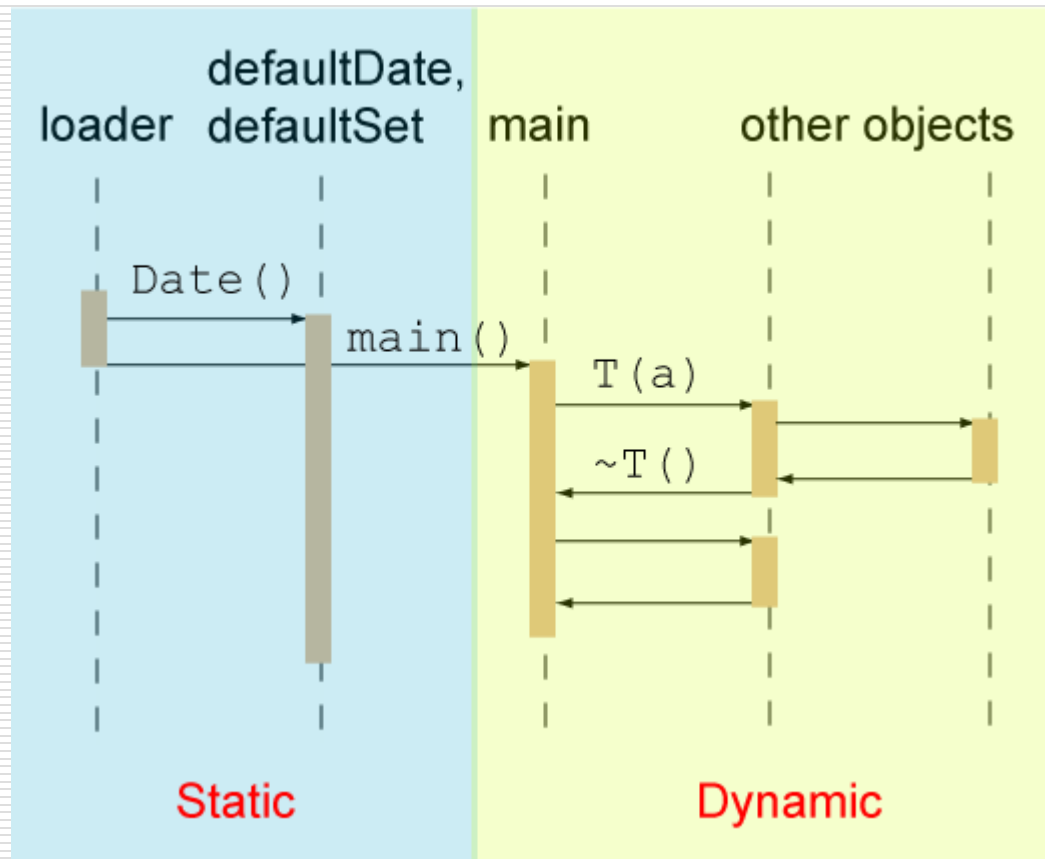
- ❑ Пряма робота з пам'яттю
- ❑ Оптимізація коду
- ❑ Обробка виняткових ситуацій
- ❑ Програмовані типи даних
- ❑ Функціональна полісемія
- ❑ Ієрархії класів і об'єктів
- ❑ Узагальнене програмування
- ❑ Стандартна бібліотека



# Приклад. Структура офісу

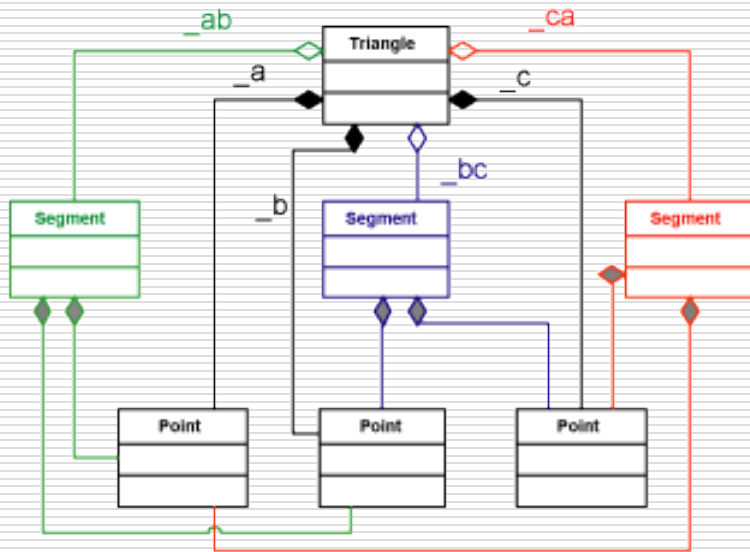


# Перегляд процесу виконання



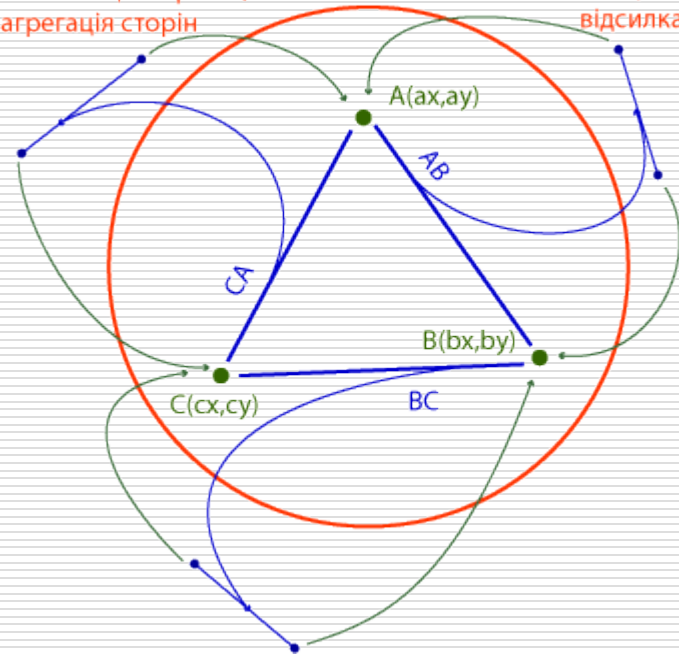


# Ієрархії об'єктів

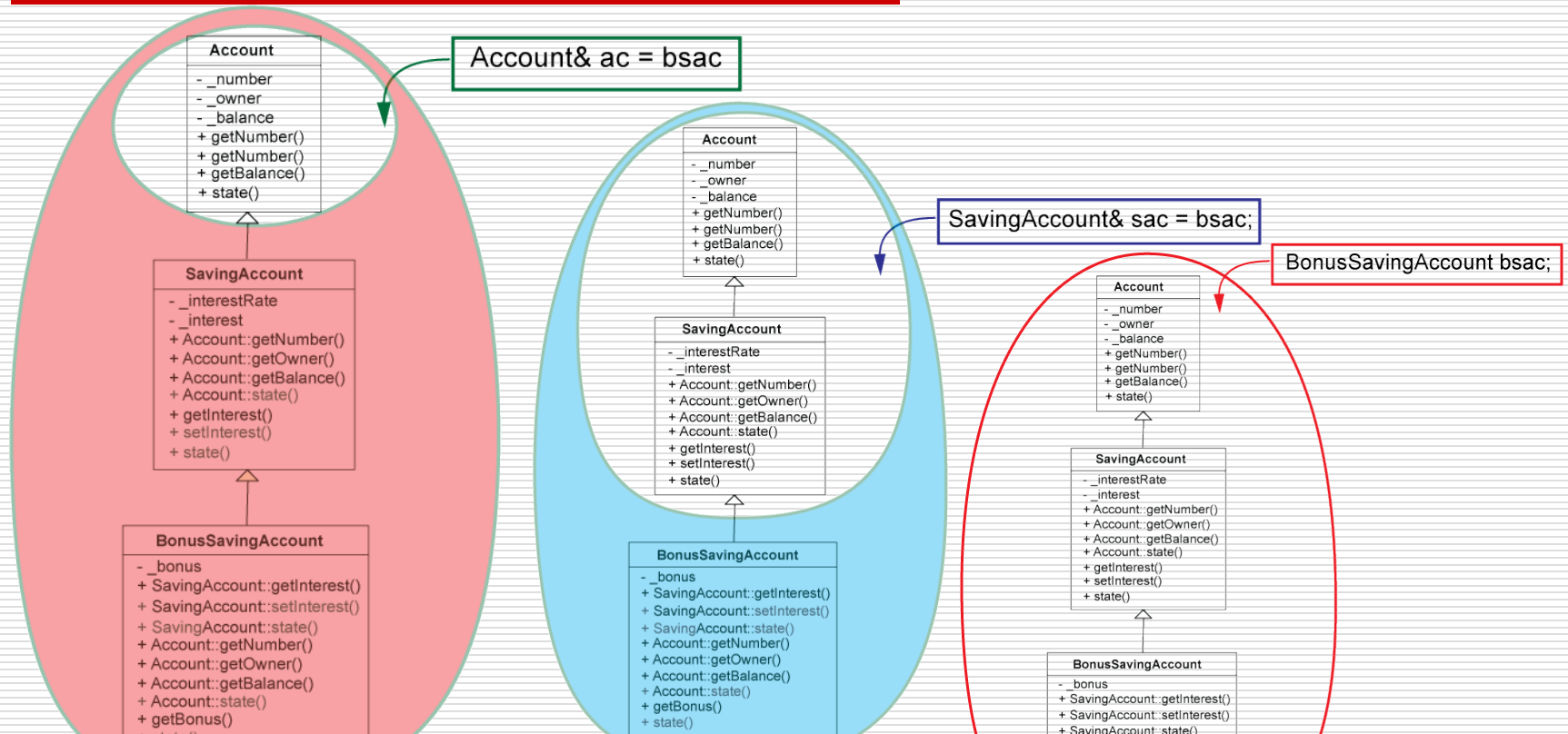


Трикутник  
композиція вершин,  
агрегація сторін

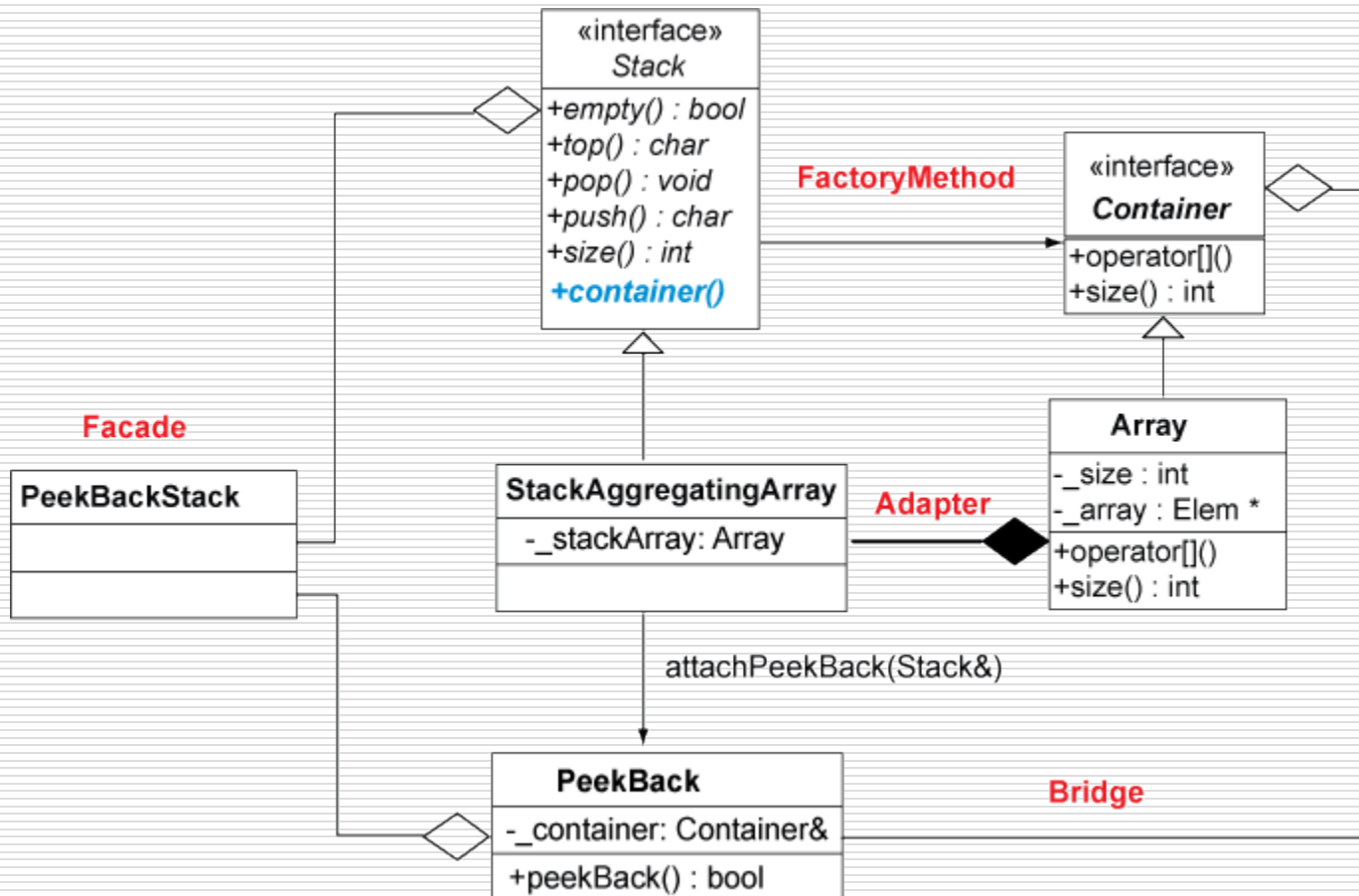
сторони агрегують  
вершини  
відсилками



# Ієрархії класів (спадкування)



# Патерни програмування



# Приклад

```
#include <iostream>
#include <cassert>
#include "quickFibonacci.h"
#include "functionsForTesting.h"

using namespace std;

//Визначимо функції
```

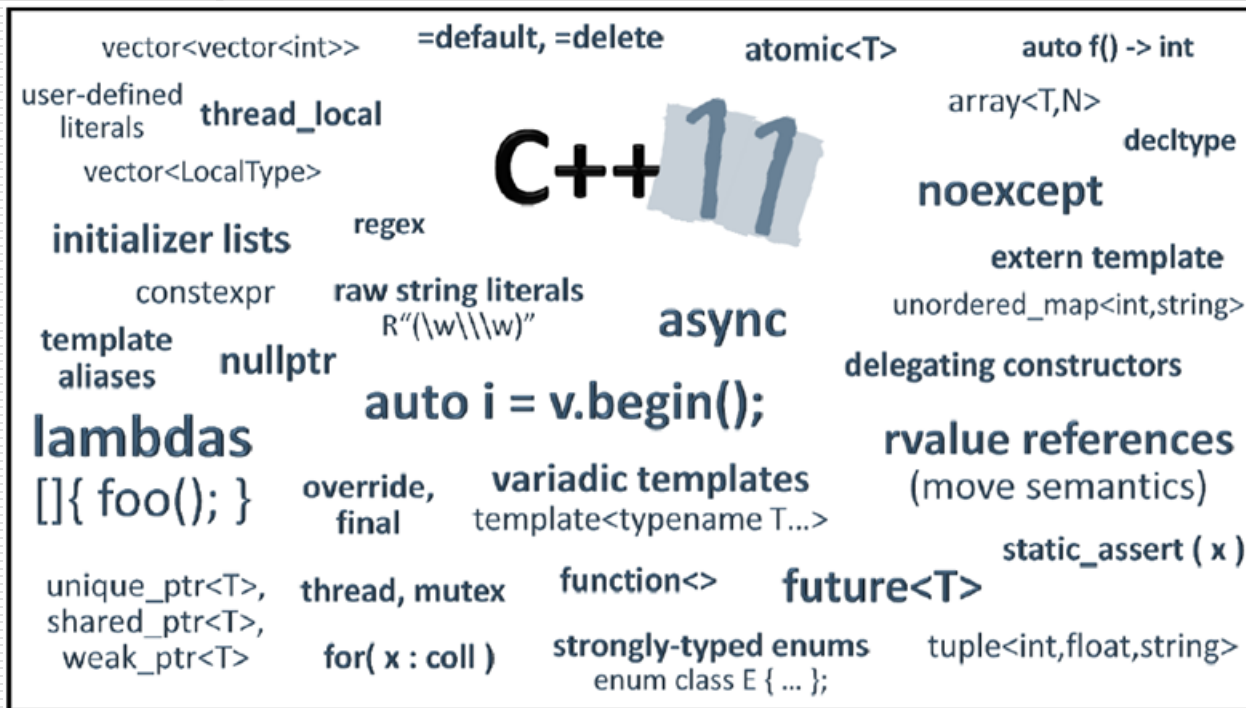
```
#include <iostream>
using namespace std;
// Допоміжна функція для піднесення до степеня
void fib(double f0, double f1, unsigned int n) {
    if (n == 0) return f0;
    if (n == 1) return f1;
    double f2 = f0 + f1;
    while (n-- > 1) {
        f0 = f1;
        f1 = f2;
        f2 = f0 + f1;
    }
    return f2;
}

int main() {
    unsigned int n;
    double f0, f1;
    cout << "Enter number of iterations: ";
    cin >> n;
    f0 = 0;
    f1 = 1;
    double result = fib(f0, f1, n);
    cout << "The number of iterations of the auxiliary function: " << result << endl;
}
```



# Що далі?

- Вивчення та дослідження нових стандартів C++11 і C++14

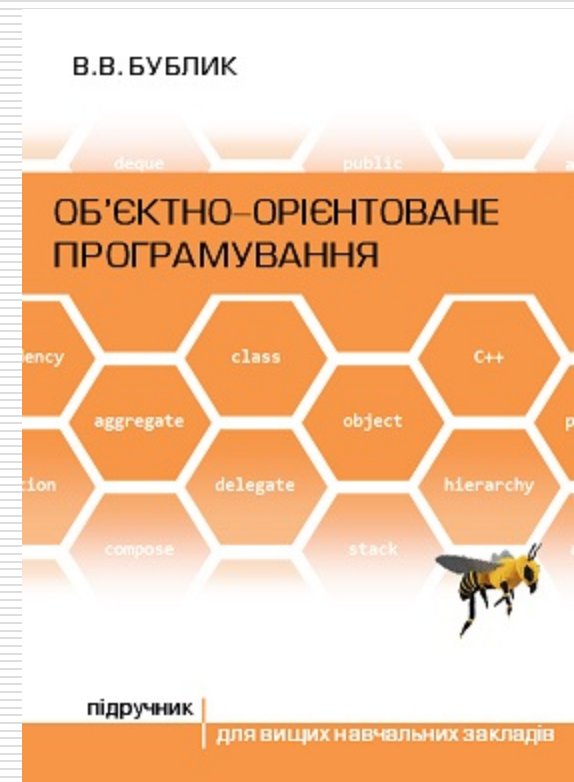




# ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ

---

Підручник для вищих  
навчальних закладів  
ISBN: 978-966-97182-1-1  
2015 р., 640 с.  
Тверда обкладинка  
[boublik@ukma.edu.ua](mailto:boublik@ukma.edu.ua)



# КУРСИ ЗА ПІДРУЧНИКОМ

---

- ❑ Процедурне програмування (осінь, 2 курс) <sup>1)2)</sup>
- ❑ Об'єктно-орієнтоване програмування (весна, 2 курс) <sup>1)2)</sup>
- ❑ Методи об'єктно-орієнтованого програмування (осінь, 3 курс) <sup>3)</sup>

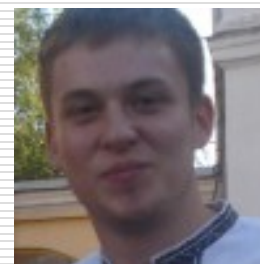
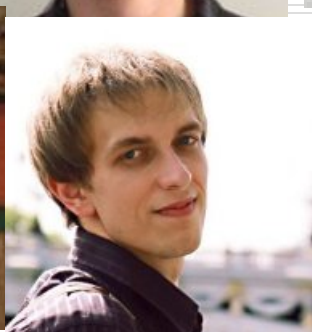
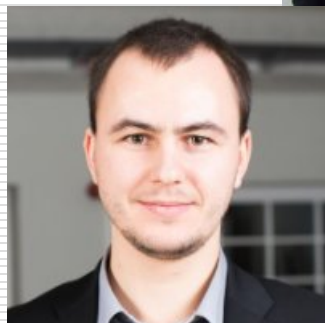
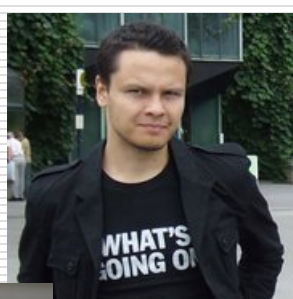
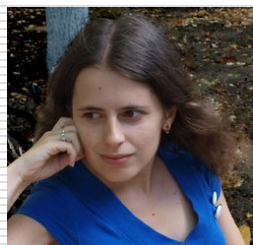
1) Нормативний для інформатики і програмної інженерії

2) Вибірковий для прикладної математики

3) Вибірковий для всіх спеціальностей

# Подяка студентам - моїм помічникам

---







**Дякую за увагу!**

☐ Вчимося програмувати так, щоб наші уміння і програми вистояли випробування часом, як Арена у Вероні (I ст. після РХ)

# Sequel

---

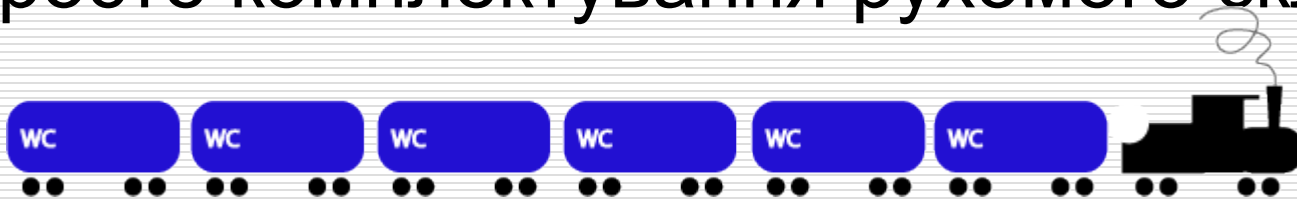
## Притча програміста (за Е.Дейкстрою)



# Притча програміста (за Е.Дейкстрою)

---

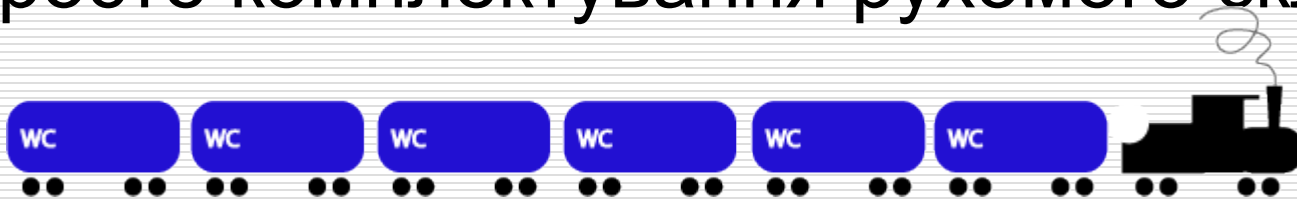
- Просте комплектування рухомого складу



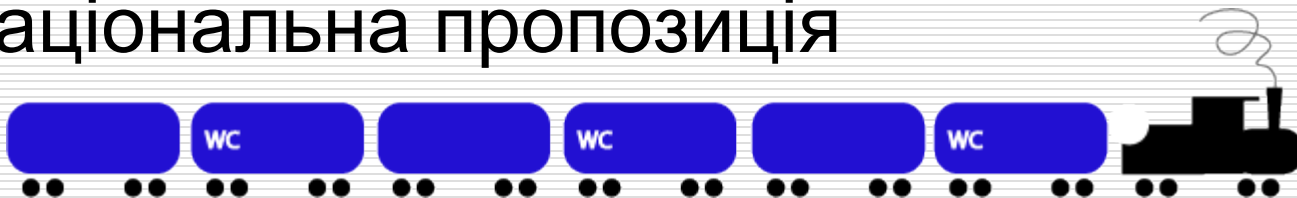
# Притча програміста (за Е.Дейкстрою)

---

- Просте комплектування рухомого складу



- Раціональна пропозиція

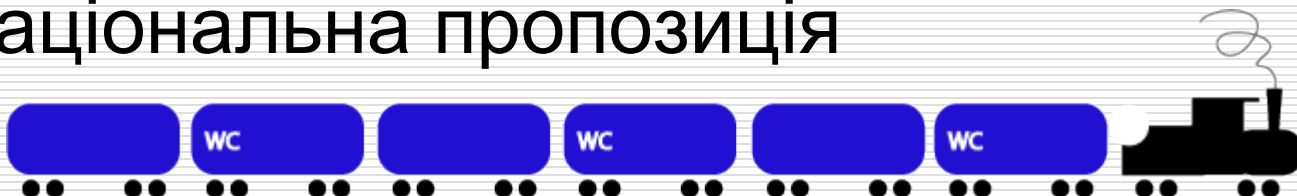


# Притча програміста (за Е.Дейкстрою)

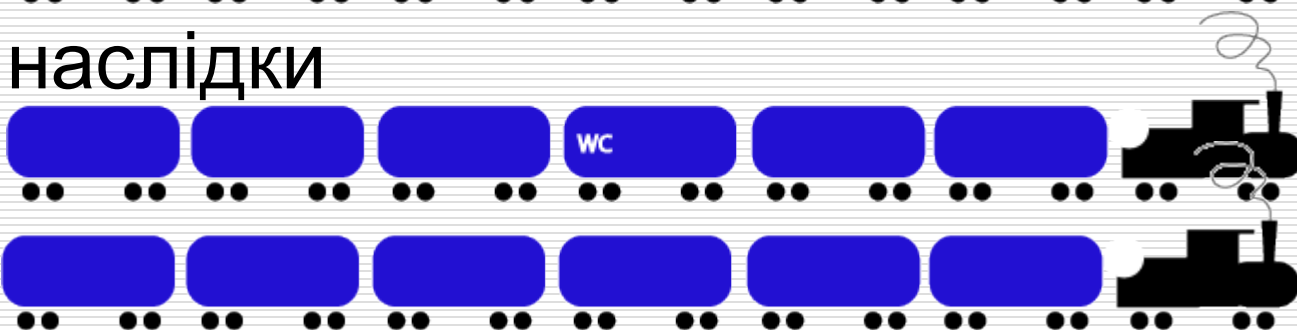
- Просте комплектування рухомого складу



- Раціональна пропозиція



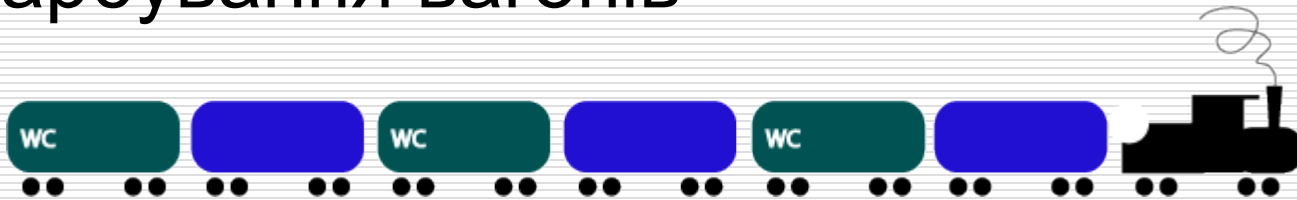
- Її наслідки



# Притча програміста (за Е.Дейкстрою)

---

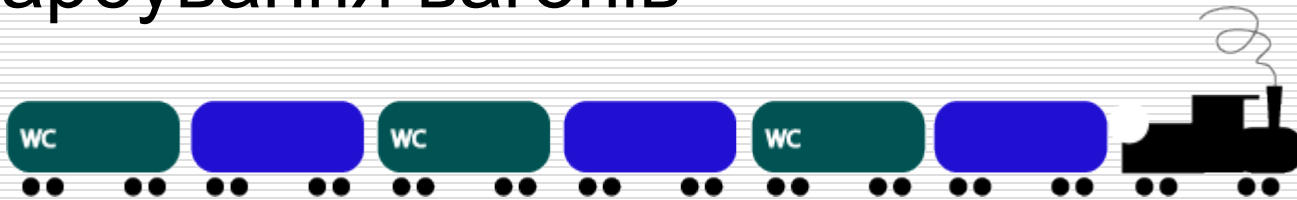
## ☐ Фарбування вагонів



# Притча програміста (за Е.Дейкстрою)

---

## ☐ Фарбування вагонів



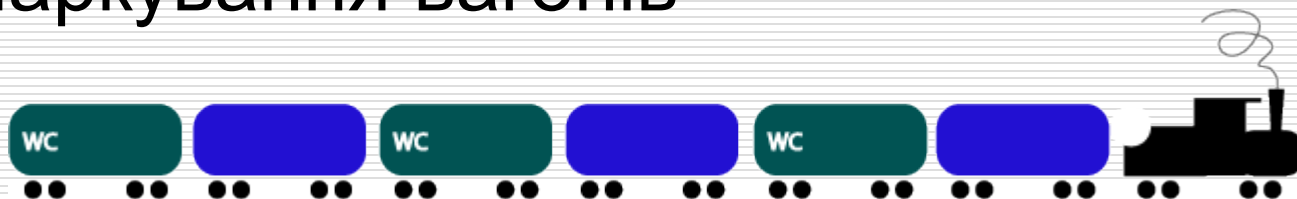
## ☐ Проблема орієнтації





# Притча програміста (за Е.Дейкстрою)

## ❑ Маркування вагонів



## ❑ Проблема орієнтації



## ❑ Маркування напрямків



# Притча програміста (за Е.Дейкстрою)

- ❑ Рішення геніального програміста:  
правильно вибрати структури даних,  
жорстко скріпивши вагони парами



- ❑ *"I have told the above story to different audiences. Programmers, as a rule, are delighted by it, and managers, invariably, get more and more annoyed as the story progresses; true mathematicians, however, fail to see the point"*

E.Dijkstra, A parable, EWD594, 1973